

# Plano de Ensino

<b>COORDENAÇÃO/CURSO:</b> Engenharia de Computação	
<b>PROFESSOR:</b> <i>Tiago Eugenio de Melo</i>	<b>PERÍODO LETIVO:</b> 2019.1

<b>DADOS SOBRE A DISCIPLINA</b>		
<b>DISCIPLINA:</b> LINGUAGEM DE PROGRAMAÇÃO I		<b>PRÉ-REQUISITO:</b> -
<b>CÓDIGO DA DISCIPLINA:</b> ESTECP001		<b>TURMA:</b> EMC01_T01
<b>CRÉDITOS</b>		<b>CARGA HORÁRIA</b>
<b>AULA</b>	<b>TRABALHO</b>	
4	0	60

<b>EMENTA</b>
Fundamentos da resolução de problemas (Lógica, Combinatória e Matemática). Construção de fluxogramas. Conceito de algoritmo. Desenvolvimento de algoritmos, variáveis, tipos de dados, constantes, operadores aritméticos, expressões, atribuição, estruturas de controle (sequência, seleção e repetição). Estruturas básica de dados (vetores e matrizes).

<b>AVALIAÇÃO</b>
<ul style="list-style-type: none"><li>• <b>Avaliações Parciais 1 e 2</b> - Resolução de problemas individualmente utilizando caneta (lápiz) e papel;<ul style="list-style-type: none"><li>o Avaliação Parcial 1 (AP1).</li><li>o Avaliação Parcial 2 (AP2).</li></ul></li><li>• <b>Mini Testes Práticos (MTP)</b> - Realização de 4 mini testes obrigatórios ao longo do semestre. Podem ser resolvidos de maneira escrita ou por meio de um programa de computador que atenda a um conjunto de testes. Ao final, será considerada a média aritmética dos mesmos no cálculo da média parcial.</li><li>• Média Parcial (MP) = <math>0.7*((AP1 + AP2) / 2) + 0.3*MT</math>;</li><li>• Se <math>MP \geq 8</math>, então o aluno está <b>aprovado</b>;</li><li>• Prova Final (PF): Avaliação escrita a ser realizada em data a ser estabelecida oportunamente;</li><li>• Média Final (MF) = <math>((2* MP + PF) / 3)</math>;</li><li>• Se <math>MF \geq 6</math>, então o aluno está <b>aprovado</b>;</li><li>• Para ser <b>aprovado</b>, o aluno precisa comparecer a, pelo menos, 75% das aulas;</li><li>• Nos demais casos, o aluno está <b>reprovado</b>.</li></ul>

# Plano de Ensino

## METODOLOGIA

Aulas expositivas em sala de aula com uso de projetor e quadro branco. Utilização da linguagem de programação Python 2.7. O ambiente para construção de programas é livre. Resolução de exercícios em sala de aula. Utilização da plataforma `run.codes`. Emissão de listas de exercícios para prática individual.

## BIBLIOGRAFIA

1. MENEZES, N. N. C. (2014) *Introdução à Programação com Python: Algoritmos e Lógica de Programação para Iniciantes*. Editora Novatec.
2. DOWNEY, A.; ELKNER, J.; MEYERS, C. (2002) *How to think like a Computer Scientist - Learning with Python*. Green Tea Press.
3. GUTTAG, J. V. (2013) *Introduction to Computation and Programming Using Python*. Editora MIT Press.
4. LOPES, G.; GARCIA, A. (2002) *Introdução à Programação*. Editora Campus.
5. SEVERANCE, C. (2013) *Python for Informatics - Exploring Information*. Editora Amazon Create Space.